Journal of Data Acquisition and Processing
ISSN: 1004-9037
https://sjcjycl.cn/
DOI: 10.5281/zenodo.777055

A NOVEL USLW BASED INTEGRATION ALGORITHM FOR OPTIMIZED DEMANDABLE RESOURCE PROVISIONING IN HYBRID CLOUD SERVICES

V. Mahavaishnavi
Research Scholar, Department of Computer Science and Engineering
Annamalai University, Annamalai Nagar – 608 002
Email:vaishnavipriya95@gmail.com

Dr. R. Saminathan
Associate Professor, Department of Computer Science and Engineering
Annamalai University, Annamalainagar – 608 002
Email: samiaucse@yahoo.com

S. Nivetha
Assistant Professor, Department of Computer Science and Engineering
School of Computing, Sathiyabama Institute of Science and Technology
Chennai - 600119, Email:nivethacse555@gmail.com

ABSTRACT

The fast spread of cloud computing, from specific customers to major businesses, has made it harder for cloud organizations to preserve information and services in the network infrastructure (cloud). Inefficient resource control has the potential to diminish cloud computational resources. As a result, resources should be distributed evenly to various stakeholders without endangering the organization's revenue or customer satisfaction. A client's requirement cannot be withheld endlessly simply because the basic resources are not available. In this paper, a novel integration algorithm named Ubiquitous Shuffled Leaf-frog with Whale Optimization (USLW) has been used to resolve the aforementioned issue within optimized procurement, dynamic allocation, and improved resource position in cloud computing. As a result, load optimization and balancing, energy efficacy, and improved resource scheduling are obtained in a hybrid-cloud model. The whale optimization algorithm surpassed many existing methods in form of response time, execution time, energy consumption and throughput in the case of multiple server settings to obtain better QoS in hybrid-cloud operation on the Cloud Service (CS) contributor end.

Index Terms - Task allocation, ubiquitous, throughput, Whale Optimization, hybrid-cloud and service provider.

I. INTRODUCTION

A spectacular development brought out by the first wave of laborer farmed in the PC industry, cloud services aids in the cessation of virtualization activities [1]. Appropriate processing is depicted as a "connection" between computing, FaaS and PaaS [2] [3]. Everyone has a unique general thesis concerning business. Purpose of appropriating calculation to build a virtualized resource of PCs, employees and experts produce that application to service the customers, pay regard to that purchased pattern [4]. Because the cloud is based on the pillars of two fundamental bases, such as cloud computing systems, it is also crucial to have access to the Internet and its architecture. Many cloud apps may use the connection for cloud services and other purposes [5]. The cloud's QoS distribution network is integrated with its features and architecture. Consequently, increasing ASP's [6] recognize that separation in among definite consumption & operate of this necessary upgrades and it used equipment lease network operators. From that instance, Forced Squares has used Amazon EC2 Analysis for 5 million hours, conserving 53% worth to satisfy quantifiable demands [7] and establishing first multiple cloud monitoring recourses.

The ASP regularly evaluates the subscription services, determines how best to allocate objectives and capabilities, and doesn't waste money on additional calculations, processing or transmitting data [8] [9]. This is especially true with regard to the demand prediction schedule. Additionally, resource distribution should be in line with decentralization since the provider may supply services that are one-of-a-kind or mixes of these methods depends on resources, compounding the problem instead of calling for complicated bids [10]. It is possible to get relevant resources from a variety of sources, and numerous users may compete for much the same services [11] [12], with providers seeking other supplier's, customer's contributions, including customer's proposals. The majority of problems in cloud applications include managing memory extension, project scheduling, power, security and data security. However, job management is often a significant area of study in cloud services. Cloud services jobs often demand high performance, ideal timeframes, quick responses and the availability of resources in terms of making advantage of such resources. The allocation plan must distribute the jobs appropriately due to diverse goals it serves.

It can serve customers utilizing the Internet [13]. Amazon has created EC2, Google Server, Apache Hardtop and Windows Azure since introducing cloud computing. A resources system cluster is what Amazon EC2 is. A Linux virtual server resource called Amazon Data Center offers web services [14]. Events may be small, big, or extremely huge [15]. [16][17] Cloud technology has an important authority on this IT business, so there is a lot of rivalry between organizations over the quality of their delivering services. Additionally, businesses are working to enhance or update their services using a variety of tools so that an increasing number of customers may sign up for cloud services [18]. Thus, allocation of resources and SLA [19], indicating user pleasure, impact service quality. But sizes and bounds should be stated and upper bounds are hard [20].

The following are the most important achievements made by this study:

- To control supply constraints in a cloud scenario, we suggested rescheduling user requests using the Whale optimization technique and searches strategy.
- To attain that there's a need specification of a virtualized environment in cloud infrastructure configurations.
- Utilizing USLW to develop expense and effective hybrid-cloud demand services.

This research article includes four major forthcoming sections. The sections are: (i) Related work section (ii) Proposed work section with USLW description section (iii) Results and interpretation section (iv) Conclusion section.

II. RELATED WORK

[21] Have provided the asset distribution strategy depending on distinct property appraising multiple SP’s as diverse valuations concurrently, hence enhancing the advantage. CSAMIISG's evaluated cost is closer to that authentic exchange cost, according to findings the recreation, but it is not precisely the same as the actual transfer price. The approach compares SP’s with IN’s. They will update the program foundation for upcoming tasks and alter the parameters to improve it. To address these challenges, [22] suggested a YARN's efforts. Dispersing progressing resources is one stage. For one-time managing the financial actions, the long-haul asset reasonableness (LTRF) paradigm is used. It provides different tiered long-haul asset reasonable (H-LTRF), with the ability of LTRF growth to incorporate advancing sources like the LTRF and H-LTRF. By introducing LTRF and H-LTRF, LTYARN topic illustrates that this results in preferred legalization of resources over the existing assessor, according to their analyses.

Computing system [23] has been introduced, but it enables asset classification, asset estimation, and a realistic sale based on consumer assessments and attributes. Clients might submit many soliciting at once depending on the payment plan. They may, however, respond to many requests, even one with an ambiguous appearance. They demonstrate that asset providers may get more social benefits and actual assistance for the organization. They provide a method for handling asset identification so that quick assignment agreements can be made, and they also improve the social benefits of cloud asset providers. The installment method measures the asset provider's interest for each customer. They dissect the agreement based on an initial concept that takes into account social assistance, execution time, resource utilization, and customers.

[24] Suggested using the VM integrating resource allotment computation to achieve energy efficiency and reduce systems administration administrative stage awareness violations, integrating DCNS pieces, the frequency of redesigns, and the duration of the transportation courses. This method efficiently reduces energy consumption, the quantity of activities, and the distance travelled throughout the transportation process to the particular cloud management. [25] Introduced a resources task control method that addresses dynamic weight and resources. It enables different sources to react to different instability caused by different barriers and adds aggregated aid to join to make sure that QoS will not provide able to make the necessary for one assist. Close testing have shown that, regardless of the methodology used to manage organizational demands, resource utilization may be enhanced by providing resources with acceptable sponsorship. This method makes sure that QoS reacts continually to unusual resource needs and adapts to common intercessions.

[26] And use a two-venture different breed’s unpredictability model to predict VM help of embedded on rigorous order management, swapping CPU and space peaks for innovative

apps and VMs. So rather than anticipating courses results, they used order line computers and NLP to decrease costs. The construction of a modular two fold deterministic model that calculates VM load, including memory and CPU resources with precision and efficacy. Select programs that enhance the CPU-VMs by many more over 5 percent and then analyze the ANFIS modeling VM, CPU utilization and storing stack using the Boeing methodology. Reorganization strategies enhanced the performance and asset utilization of VM's, according to extensive testing.

[27] Offer an included method of MEC and Cloud Services on the transportation network. Cloud-MEC download outcome synchronize upload results and save up system requirements. They suggest obtaining the capacity planning software computation and converting to the CCORAO programme linked to the solution if the issue develops and the NPP grows complex. The technique efficiently reduces computer consumption and calculation time, particularly when MEC servers are unable to satisfy the demands due to a lack of processing power. There are two approaches to power consumption that [28] has presented in order to balance energy consumed with load delay: the XCS and the BCM-XCS. Our tests demonstrate the benefits of BCM-XCS over standard XCS approach. Cloud and fog node load dispersion reduces computing and management latency. Controlling processor variations has the primary benefit of steadily reducing latency by 42% while consuming a fixed amount of energy.

Utilizing GA and SVR, [29] created an improving functional for reconfiguration suggestions (SVR). This executive information system evaluates spare time and recommends a realistic and practicable cloud services architecture. Results revealed that predicted period were comparable to real times, effectively forecasting period, expenses and their reductions. To improve solitary energy savings, [30] research on workload and allocation of resources on a densely operating C-Ron MEC have indeed been conducted. A hybrid memory non-software application designed to improve the efficacy of burden loading, distribution of resources and administration. The Lebanon proposal divides the subject into four distinct using central variation and adaptive activities. They evaluate operational delays against renewable energy. Complex simulation demonstrates that system factors impact energy effectiveness and service interruption. The results obtained revealed the advantages of a difficult C-RON load as well as volume assessment difficulty.

[31] Suggested an energy-saving strategy for resource allocation. The forecasting systems and dynamic channel update algorithm were used to allocate resources for job execution and response time. This method reduces energy structure by lowering data centre use. The information database updating method gives accurate values back. Planning process and a decrease in energy use are excellent ways to allocate resources. [32] Proposed a one-of-a-kind secure buffering paradigm that is able to operate on a Bluetooth 6G networks for the purpose of BD on building automation (SBs). The sampling element combines IoT features alongside those of CC, EC and Big Data. In attempt to provide customers with either a secure and far more dependable environment for using the World Wide Web, exchanging and maintaining massive amounts of data in the data center, they developed a secured Caching Decision System (CDS) on such a mobile phone which operates through an SB.

[33] Developed Integrating Federation Model (InFeMo) to merge current cloud modeling with a computational intelligence scenario and other relevant technologies for a unique integrated scenario. The proposed strategy intended continue providing customers a somewhat more resource framework and atmosphere. Using CSPs' infrastructure and the PaaS paradigm to process customer request more efficiently and quickly was the preferred solution. Their research sought to fill an information need in the field of federation distributed systems. Table 1 gives a review of current studies on allocation of resources in cloud environments and related approach to problem-solving.

Table 1 An overview of relevant terms on resource distribution in cloud technology

Ref. No	Title	Proposed Explanation	Platform	Problem
[21]	Imperfect information dynamic Stack elberg game based resource allocation using hidden Markov for cloud computing	CSAMIISG's evaluated price is close to the true exchange price, but not precisely.	Huawei	Adjust parameters and the runtime environment to improve effectiveness
[22]	Fair resource allocation for data-intensive computing in the cloud	The approach gives multiple levels long haul resource reasonable (Hybrid Shuffled Leapfrog Algorithm) with the possibility of the LTRF extension to include progressing resources, for illustration, the LTRF and H-LTRF.	Amazon EC2	LTYARN open source not effective
				http://sourceforge.net/ projects/ltyarn/ (accessed on 22 August 2021)

[23]	An online auction mechanism for cloud computing resource allocation and pricing based on user evaluation and cost	The OVRAP algorithm is suggested by the researcher.	IBM CPLEX12	C++ used for both implementation and algorithm
[24]	Self-adaptive resource allocation for energy-aware virtual machine placement in a dynamic computing cloud	The suggested procedure initially groups the systems with the shortest route lengths using the provided DCN topology.	Google cluster Track	lacks a lot of useful information
[26]	ANFIS with natural language processing and gray relational analysis based cloud computing framework for real-time energy-efficient resource allocation	ANFIS model handles activities are aimed estimation by retraining feature characteristics.	Malleable NS2	lacks a lot of useful information

III. PROPOSED WORK

The suggested study offered a hybrid optimization and classification approach for dynamically allocating resources in a scenario. This hybridization strategy includes the SLFA, the UBS and the fundamentals of genomes. Shuffled leaf frog approach examines and characterizes all accessible multi-cloud characteristics from the shared database. The commodities that come from these very small groups are chosen as a result of demand and requirement. The basic requirement of every binary code is met with capabilities depending on need. To perform a specific operation on a particular VM instances, customer resources will be required and thorough cost and utilization optimization is essential. The temporal resource may be taken from VM in same pools or common pools. After the task is finished, these borrowed resources will be handed away. The suggested method made sure that demand-based resources were used as effectively and cheaply as possible.

The limited number of resources developed on variables like productivity, efficiency and cost as a result of energy efficient manner. The best and most trustworthy supply pool must be found, nevertheless, using two distinct search techniques: internal and external searched. This method achieves the greatest speed and least cost to scan from all around given Virtual machine in a several cloud scenario. The precise interaction between the SLFA and UBS algorithms to achieve efficiency was shown in the chart in Figure 1.

The specific Virtual Machine produces requests for assets from a shared database using below.

$$VR_t (y) = \{vR_t 1, vR_t 2, vR_t 3, \ldots, vR_t n\} \quad (1)$$

Where $VR_t (y)$ represents the total of service needs of whatever Virtual Machine as well as y could be anything authentic value among 1 - n. based on a real-world situation and the UBS search algorithm, the request created. The link between optimizing and classifications of the suggested dynamic resource allocation approach is elaborated in the diagram shown in Figure 1.

Fusion of Optimizer and Classifier

The implementation of SLFA and UBS in several clouds is necessary for hybrid efficiency. By combining these two, a common pool of resources may be allocated dynamically at their best [34]. Comparing the hybrid installation to earlier scheduling and scaling strategies, it offers the most optimal option. This approach offered an implementation of Shuffled Leapfrog Algorithm**,** Unified Binary Search, Genetic Algorithm and hybrid Shuffled Leapfrog Algorithm throughout this part.

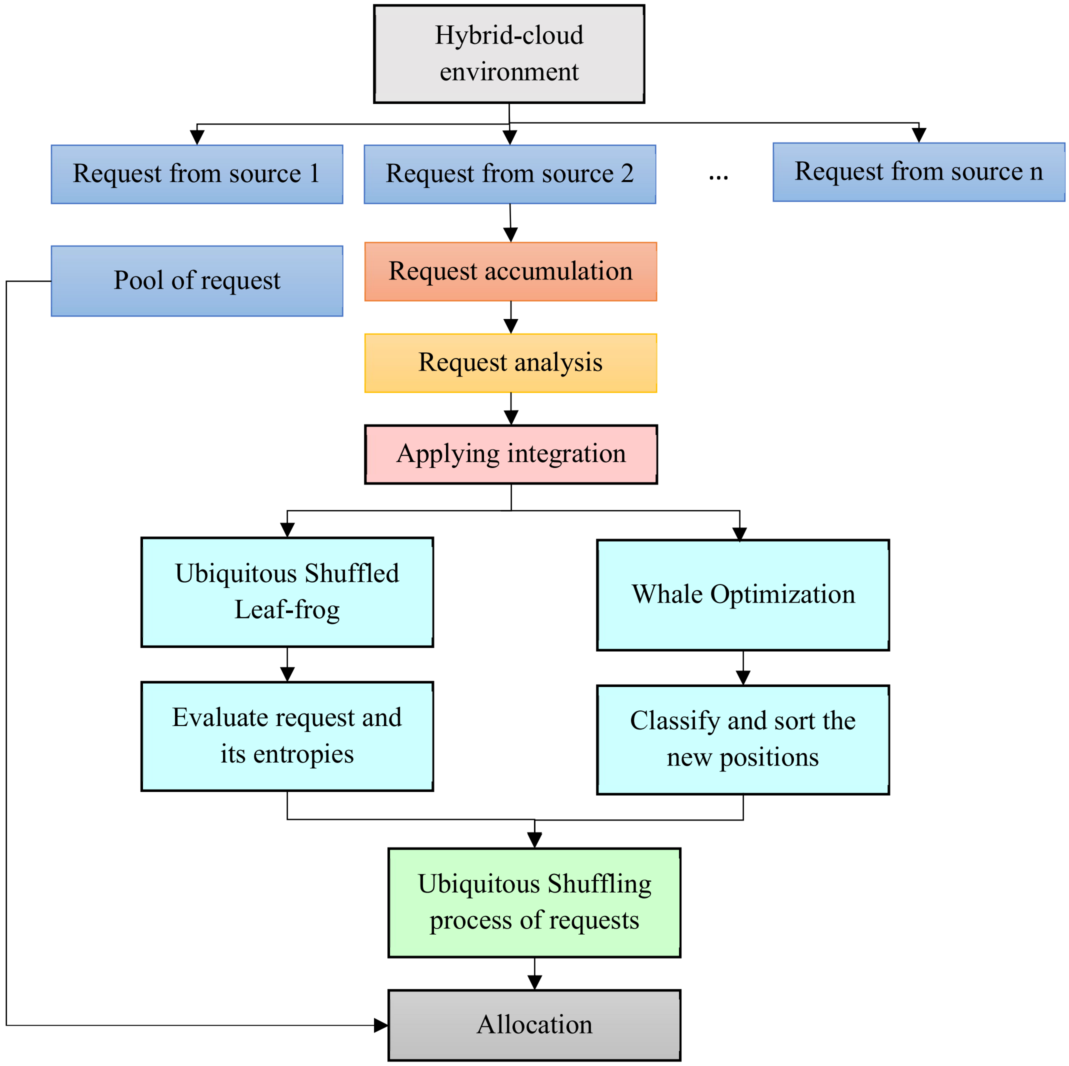

Figure 1 UBS and SLFA Optimization Construction

Ubiquitous Shuffled Leapfrog Algorithm (USLA)

Knowing that total variety of groups and the total amount of wealth within every category is necessary when creating subgroups. If the numbers of groups and also the quantity of each kind of material within every collective are treated like one, the amount of funding are T = G*R. The functional form of each unit with the greatest value determines which groups have the best resource usage. The total resources are separated into tiny groups; for example, each kind of commodity must belong to a subgroup to meet need-based distribution. Choose the resource's initial connection first from main section and secondly participation from the third group. Add the findings to G. After the divide is accomplished, a certain quantity of data streams will continue [35].

External Research

Metadata exploration is separated into external and internal steps. This stage includes choosing the best pair of resources to satisfy the requested task.

Stage 1: Cases: choose H and S, where H stands for a condensed range of capabilities and S for the total quantity of resources. Thus, there will be a total lot of organizations with all necessary resources T = H*S.

Stage 2: Replicate virtual groups: Using the information at hand, a sample of R virtualization will be vert (1), vert (2) …. vert (s).

$V(i) = \{vi_1, vi_2, \ldots, vi_d\}$ where D presents the issue of choosing a specific component within the group.

Stage 3: Sectioning and sorting: Arrange the resources in a specific group in order of decreasing. The whole crew will indeed be vert (i), src (i) and i = 1 … to s. position of (vert =Px(i)).

Stage 4: Grouping of materials: Split collection X axis into Y axis such that each receives N resources.

Stage 5: Input parameters with each group: Each Y_x for each category, wherein x = 1, 2, 3, and so on, was evaluated using the internal search indicated below.

Stage 6: Aggregation of groups: Every group comprises (Y_1... Y_g) the quantity of every sort of material for a certain pool after assessment.

Stage 7: If the convergent requirements are satisfied, go to the fourth stage of query and analysis; otherwise, cease.

Internal Research

The external search's fifth stage included N separate selections of groups. After searching, the method computes to global research. The stages for questionnaires and surveys were as followed.

Stage 1: Locate iGr and iNode to zero. Where, iGr be the group and iN be the progression steps.

Stage 2: 1 + iGr = iGr

Stage 3: 1 + iNode =iNode

Stage 4: Subgroup formation: Formation of groups where greater value is mainly reflected one or lower value is connected with a lower one. Using a triangle posterior distribution, value is assigned.

Stage 5: The worst location is corrected by using the minimum especially prevalent and selecting leap parameters if the commodity, including RAM, has a different composition than necessary. Optimally, proceed to Stage 8 of the research process if the amount has already been reallocated to a particular item; alternatively, proceed to Step six of both the exterior searching.

Stage 6: Given the highest value supplied, determine the resource's size. If findings from Stage 5 don't improve things, efficiencies may be evaluated utilizing the size of the specific item. If a resource's effectiveness is higher than that of the previous one after it has been given a new value, the value of the prior resource may be used instead. If not, continue with the internal design process.

Stage 7: If the updated value of something like the resources does not satisfy the specified demand, then a virtual computer resource will be produced at randomness and added to the pool, replacing the original.

Stage 8: After replacing the cheapest component with a synthetic one, sort the collection using the ULS technique.

Stage 9: If N > iN Replicate Stage 3 of the Internally Searching.

Stage 10: If G > iG Internal search returns to the ongoing search if the subgroups cannot be combined in the first phase of internal process.

Ubiquitous Search

Everyone here at US is aware of both the intricate algorithm for binary searching utilized to locate that required data. Binary search algorithms usually use a consecutive search area, making result exceedingly expensive and difficult to get. The difficulty of the search situation is greatly impacted by the algorithm's cost. With any chronology and using any media or device, US may search anywhere, at any time. Therefore, its use in modern decentralized system, such as multi-cloud, is particularly beneficial. In our suggested paradigm, US are used without loops and without an equal check. For example, employing = and >= to determine the needed output is highly beneficial in reducing reviewing and looping and increasing computational efficiency at a cheap cost. The optimum option in low will always be smaller than high and a midpoint when calculating the median using the lowest and highest technique or resources. The technique will indeed be log-cost (n).

Algorithm 1: US formulation

Define US(a, key) # a is the arr and attributes is the range were desire to searching
log= 0
bi = a.len-1
while(bi-lo>1)
middle = lo + (bi-lo)/2
if a[mid]<=key
log=middle
else
bi=middle
end
if (a[log]== key)
go back log
else
return "rate not found"
end all

Whale Optimization Algorithm (WOA)

Its ability to locate their target before engulfing them. WOA approach presupposes that now the latest professional up-and-coming combination is the target setting or is near to the optimal frequency. Since the location of the effect is obtained in the chase space is not known in advance. This is because the pursuit space is not known. When a best representation is identified, others will attempt to modify its qualities to align with the best. [35] Have provided an explanation of the mathematical formula. Figure 2 below explains this procedure.

Algorithm 2: WOA pseudo code

Begin the whale populations at randomly.
Analyze the fitness levels of whales to choose the most effective search strategy Y*
While $z < z_{max}$
Determine the objective functions for every agents
Every searching mechanism
If b < 0.5 where h is the random number between 0 and 1 then
If$\lvert\vec{C}\rvert$< 1 then
$Y(z+1) = \overrightarrow{Y*}(z) - \vec{C}.\overrightarrow{D^n}$ (2)
Else if $\lvert A \rvert \geq 1$ then
$Y(\vec{z}+1) = \overrightarrow{Y_{rand}}(z) - \vec{C}.\overrightarrow{D^n}$ (3)
end If
Else If b ≥0.5 then
$Y(\vec{z}+1) = \overrightarrow{D'}.e^{bl}.\cos(2\pi l) + \overrightarrow{Y*}(z)$ (4)
End If
End For
Estimate the condition of Y(z + 1) and inform $Y^{/}$
End While

In Whale Optimization method, b is the randomly numbers range 0 and 1. In Equation (2), Y(z + 1) is the simplified situation, Y* (z) is the most excellent resolution and techniques C = (2*a*r)-a where a is linearly reduced between 2 to 0 approached on maximum iterations number reduction encircles and r is ran domed vectors range from [0,1] and $D = |(2*r*Y^*(z)) - (Y(z))|$ where Y (z) is vector. In Equation (3), $D = |(2*r*Y_{rand}(z)) - (Y(z))|$ and $Y_{rand}(z)$ vector. In Equation (4), describes the log spiraled shapes, L is randomized number of limits -1 and 1 then $D = Y^*(z) - Y(z)$.

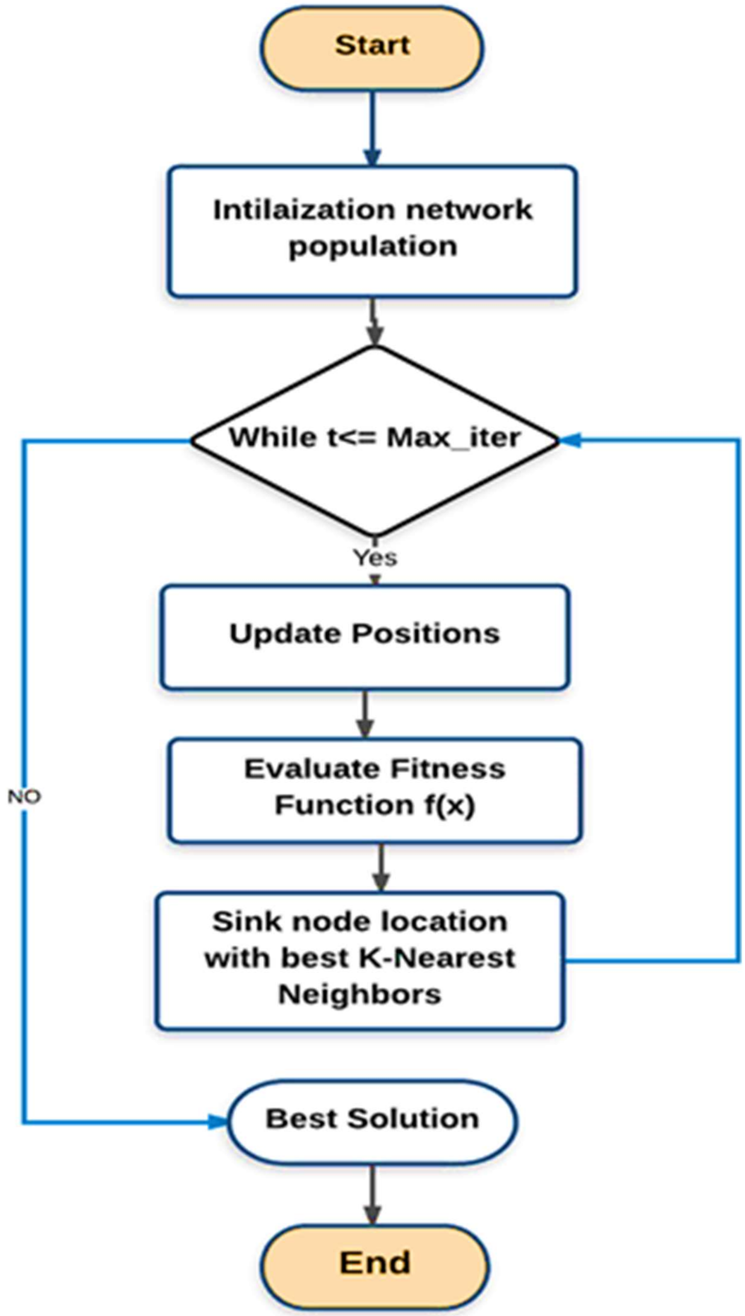

Figure 2 Whale optimization method Flowchart

Toolkit of Simulation

Analyzer is a simulator toolkit that offers reproduction of this essential features cloud, including occasion meeting out, the formation of fog organizations, inter-entity communications, etc. This toolkit offers a lot of options and services, including:

- Test the functionality of programmes in a regular, regulated setting.
- Runs tests using a variety of workload combinations and performance circumstances on a development oriented infrastructure model.
- Tune the boundary condition before deploying applications in a real cloud.

- It is an all inclusive framework for modeling the provision, deployment, as trade contractors of the cloud.
- It offers a cloud environment simulation that can link data both the public and private sectors.
- Virtualization engine capability for creating and managing separate virtual operations on a DC node.

IV. NUMERICAL RESULTS

Balancing of Load

Table 2 compares the Time of Response, Time of Execution and Consumption of Energy usage of the UBS, BAT, WOA and PROPOSED systems that have been used in different study stages. Testing was done on the load balanced, performance, and reaction times of each algorithm and the findings reveal that although the utilization is the same for all algorithms is 100ms. The response times for PSO, CSO, CSA and WOA are all zero milliseconds. Automation takes 248.38 ms to execute UBS, 103.633 ms to execute BAT, 101.634 ms to execute BAT and 90.7289 ms to execute WOA. PSO uses 22,355.147 Joules of energy, BAT uses 9322.228 Joules, WAO uses 8165.604 Joules, and all other systems use a combined total of 9147.009 Joules. Thus, it has been discovered that when comparing the whale optimization approach to the USB, BAT and WOA algorithms for congestion control, bandwidth allocation, and energy efficiency, it produces the best overall performance. In real time, schedule length and energy demand. Suggested outperformed some other techniques evaluated in this study.

Table 2 Comparisons between the Algorithms

Algorithms	Parameters		
	Time of Response	Time of Execution	Consumption of Energy
UBS	0.1165	248.38	22,355.147
BAT	2.096	103.581	9322.228
WOA	0	101.634	9147.009
PROPOSED	0	90.7288	8165.604

Energy Consumption

The energy consumption of the UBS, BAT, WOA and PROPOSED algorithms while several servers are active is compared graphically in Figure 3. It demonstrates that the WOA algorithm consumes 8165.604 Joules of energy, whilst PROPOSED produces results that are 4% better than BAT. PROPOSED 12% improved. Making PROPOSED 13.22% better. Since UBS uses 22355.147 Joules of energy, PROPOSED is 92 times higher than UBS. Therefore, we may conclude that PROPOSED is superior to all other strategies in terms of its energy use.

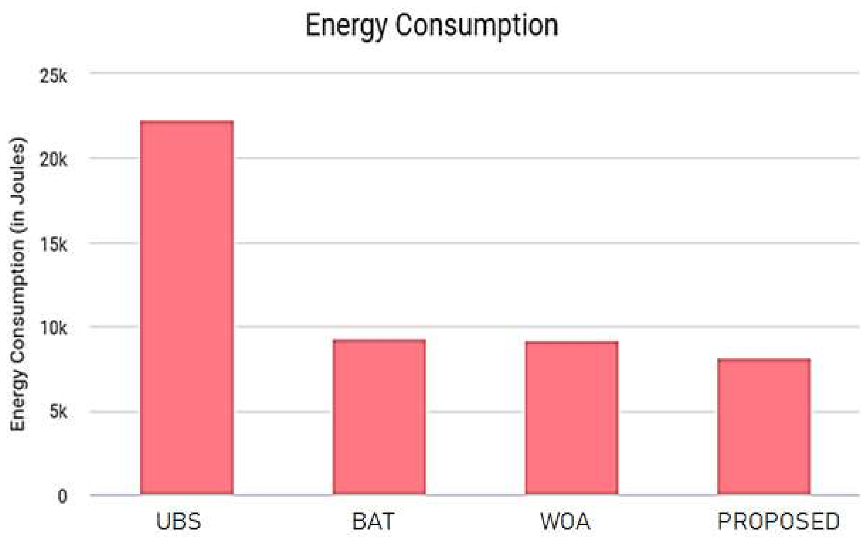

Figure 3 Comparisons between the Proposed Method and the Energy Consumption of the UBS, BAT and WOA

Execution and response time

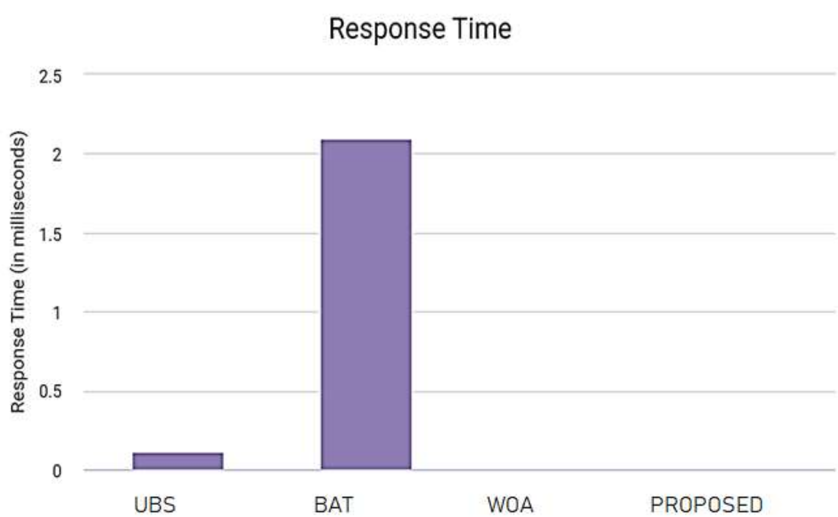

Figure 4 Comparison between the Proposed Method and the response time of the UBS, BAT and WOA

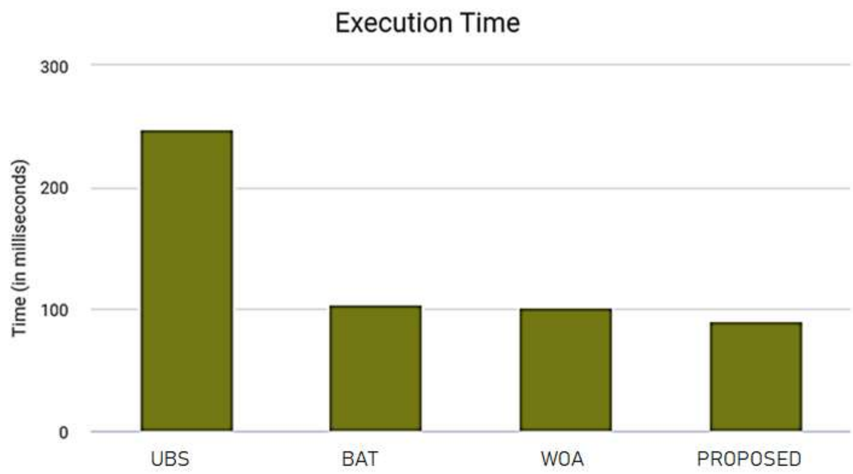

Figure 5 Comparisons between the Proposed Method and the Execution Time of the UBS, BAT and WOA

Figure 5 compares the execution speed of and PROPOSED. The findings indicate that PROPOSED provides 4 percent better query execution outcomes than BAT, with WOA taking 90.7288 ms and BAT taking 94.406 ms. Similar to how WOA is 11% faster than the BAT algorithm when comparing job execution times, BAT takes 101.633 ms. WOA is contrasted with CSA and UBS as well. WOA performs 92 percent better than PSO and is 13% better than CSA. Task execution times for CSA and PSO are 103.580 ms and 248.390 ms, respectively. Thus, we can draw a conclusion that WOA requires shorter execution time than the other methods. Various servers with different speeds, RAM, and bandwidth are taken into consideration for the testing environment with varying numbers of jobs. The findings indicate that the WOA among all the algorithms, algorithm produces the best results. When WOA is implemented, cloud platforms consume much less energy and response and execution times are also reduced when compared to other methods. WOA is more cost effective than other methodologies because it requires less energy and takes less time to respond and execute.

V. CONCLUSION

During multi-cloud implementation, adaptive distribution of resources and optimal energy options for specific virtualization to satisfy demand fossil fuel consumption is a significant challenge for providers of cloud-based services. Multi-cloud architectures struggle with rushed effective interventions. USLW's solution is used to optimize and save resources in the deployment of customer operations from a central registry in a separate architecture. Suggested method implementation in cloud analyst for simulation and data extraction results shows that compared to a single algorithm, the factors taken into account provide a superior trade-off with suggested approach. The turnaround and execution times varied by just a few seconds and remained almost consistent as the number of jobs increased. One disadvantage of the current study is that we have not compared PROPOSED using cost, make span, etc. We will contrast several methods with distinct parameters choices in next study. We may use a variety of meta-heuristics strategies to enhance our job. Second, we can utilize various kinds of computing environments like fog computing and the platform as a service to spread out the work and use less energy. Our methodology can be applied to a genuine cloud provider, such as Amazon Web services, Windows Azure, etc.

REFERENCES

1. Papagianni, C.; Leivadeas, A.; Papavassiliou, S.; Maglaris, V.; Cervello-Pastor, C.; Monje, A. On the optimal allo-cation of virtual resources in cloud computing networks. IEEE Trans. Comput. 2013, 62, 1060–1071.

2. Kaewpuang, R.; Niyato, D.; Wang, P.; Hossain, E. A Framework for Cooperative Resource Management in Mobile Cloud Computing. IEEE J. Sel. Areas Commun. 2013, 31, 2685–2700.

3. Xiao, Z.; Song, W.; Chen, Q. Dynamic resource allocation using virtual machines for cloud computing environ-ment. IEEE Trans. Parallel Distrib. Syst. 2012, 24, 1107–1117.

4. Warneke, D.; Kao, O. Exploiting Dynamic Resource Allocation for Efficient Parallel Data Processing in the Cloud. IEEE Trans. Parallel Distrib. Syst. 2011, 22, 985–997.

5. Son, S.; Jung, G.; Jun, S.C. An SLA-based cloud computing that facilitates resource allocation in the distributed data centers of a cloud provider. J. Supercomput. 2013, 64, 606–637.

6. Wei, G.; Vasilakos, A.V.; Zheng, Y.; Xiong, N. A game-theoretic method of fair resource allocation for cloud com-puting services. J. Supercomput. 2010, 54, 252–269.

7. Laili, Y.; Tao, F.; Zhang, L.; Sarker, B.R. A study of optimal allocation of computing resources in cloud manufacturing systems. Int. J. Adv. Manuf. Technol. 2012, 63, 671–690.

8. Buyya, R.; Yeo, C.S.; Venugopal, S.; Broberg, J.; Brandic, I. Cloud computing and emerging IT platforms: Vision, hype, and reality for delivering computing as the 5th utility. Futur. Gener. Comput. Syst. 2009, 25, 599–616.

9. Almeida, J.; Almeida, V.; Ardagna, D.; Cunha, Í.; Francalanci, C.; Trubian, M. Joint admission control and resource allocation in virtualized servers. J. Parallel Distrib. Comput. 2010, 70, 344–362.

10. Beloglazov, A.; Abawajy, J.; Buyya, R. Energy-aware resource allocation heuristics for efficient management of data centers for Cloud computing. Futur. Gener. Comput. Syst. 2012, 28, 755–768.

11. Nathani, A.; Chaudhary, S.; Somani, G. Policy based resource allocation in IaaS cloud. Futur. Gener. Comput. Syst. 2012, 28, 94–103.

12. Lin, C. A Novel College Network Resource Management Method using Cloud Computing. Phys. Procedia 2012, 24, 2293–2297.

13. Mei, H.; Wang, K.; Yang, K. Multi-Layer Cloud-RANWith Cooperative Resource Allocations for Low-Latency Computing and Communication Services. IEEE Access 2017, 5, 19023–19032.

14. Salhaoui, M.; Guerrero-González, A.; Arioua, M.; Ortiz, F.J.; El Oualkadi, A.; Torregrosa, C.L. Smart industrial iot monitoring and control system based on UAV and cloud computing applied to a concrete plant. Sensors 2019, 19, 3316. [PubMed]
15. Khasnabish, J.N.; Mithani, M.F.; Rao, S. Tier-Centric Resource Allocation in Multi-Tier Cloud Systems. IEEE Trans. Cloud Comput. 2015, 5, 576–589.

16. Bal, P.K.; Pradhan, S.K. Privacy Preserving Secure Data Storage scheme based on Adaptive ANN and Homomorphic Re- Encryption Algorithm for Cloud. In Proceedings of the 2019 International Conference on Intelligent Computing and Remote Sensing (ICICRS), Bhubaneswar, India, 19–20 July 2019.

17. Oláh J, J.; Aburumman, N.; Popp, J.; Khan, M.A.; Haddad, H.; Kitukutha, N. Impact of Industry 4.0 on environmental sustainability. Sustainability 2020, 12, 4674.

18. Bal, P.K.; Pradhan, S.K. Multi-level authentication-based secure aware data transaction on cloud using cyclic shift transposition algorithm. In Advances in Intelligent Computing and Communication; Springer: Singapore, 2020.

19. Das, T.K.; Tripathy, A.K.; Srinivasan, K. A Smart Trolley for Smart Shopping. In Proceedings of the 2020 International Conference on System, Computation, Automation and Networking (ICSCAN), Pondicherry, India, 3–4 July 2020.

20. Tafsiri, S.A.; Yousefi, S. Combinatorial double auction-based resource allocation mechanism in cloud computing market. J. Syst. Softw. 2018, 137, 322–334.

21. Wei,W.; Fan, X.; Song, H.; Fan, X.; Yang, J. Imperfect information dynamic stackelberg game based resource allo-cation using hidden Markov for cloud computing. IEEE Trans. Serv. Comput. 2016, 11, 78–89.
22. Tang, S.; Lee, B.-S.; He, B. Fair Resource Allocation for Data-Intensive Computing in the Cloud. IEEE Trans. Serv. Comput. 2016, 11, 20–33.

23. Zhang, J.; Xie, N.; Zhang, X.; Li, W. An online auction mechanism for cloud computing resource allocation and pricing based on user evaluation and cost. Futur. Gener. Comput. Syst. 2018, 89, 286–299.

24. Jiang, H.-P.; Chen,W.-M. Self-adaptive resource allocation for energy-aware virtual machine placement in dynamic computing cloud. J. Netw. Comput. Appl. 2018, 120, 119–129.

25. Gong, S.; Yin, B.; Zheng, Z.; Cai, K.-Y. Adaptive Multivariable Control for Multiple Resource Allocation of Service-Based Systems in Cloud Computing. IEEE Access 2019, 7, 13817–13831.

26. Wu, X.; Wang, H.; Wei, D.; Shi, M. ANFIS with natural language processing and gray relational analysis based cloud computing framework for real time energy efficient resource allocation. Comput. Commun. 2019, 150, 122–130.

27. Preethi, P., & Asokan, R. (2021). Modelling LSUTE: PKE Schemes for Safeguarding Electronic Healthcare Records Over Cloud Communication Environment. Wireless Personal Communications, 117(4), 2695-2711.

28. Abbasi, M.; Yaghoobikia, M.; Rafiee, M.; Jolfaei, A.; Khosravi, M.R. Efficient resource management and workload allocation in fog–cloud computing paradigm in IoT using learning classifier systems. Comput. Commun. 2020, 153, 217–228.

29. Reis, T.; Teixeira, M.; Almeida, J.; Paiva, A. A Recommender for Resource Allocation in Compute Clouds Using Genetic Algorithms and SVR. IEEE Lat. Am. Trans. 2020, 18, 1049–1056.

30. Zhang, Q.; Gui, L.; Hou, F.; Chen, J.; Zhu, S.; Tian, F. Dynamic Task Offloading and Resource Allocation for Mo-bile-Edge Computing in Dense Cloud RAN. IEEE Internet Things J. 2020, 7, 3282–3299.

31. Praveenchandar, J.; Tamilarasi, A. Dynamic resource allocation with optimized task scheduling and improved power management in cloud computing. J. Ambient. Intell. Humaniz. Comput. 2021, 12, 4147–4159.

32. Christos, L.; Stergiou, K.; Psannis, E.; Gupta, B.B. IoT-based Big Data secure management in the Fog over a 6G Wireless Network. IEEE Internet Things J. 2021, 8, 5164–5171.

33. Stergiou, C.L.; Psannis, K.E.; Gupta, B.B. InFeMo: Flexible Big Data Management Through a Federated Cloud System. ACM Trans. Internet Technol. 2022, 22, 1–22.

34. Preethi, P., Asokan, R., Thillaiarasu, N., & Saravanan, T. (2021). An effective digit recognition model using enhanced convolutional neural network based chaotic grey wolf optimization. Journal of Intelligent & Fuzzy Systems, (Preprint), 1-11.

35. Misener, P. (2020). Food Insecurity and College Athletes: A Study on Food Insecurity/Hunger among Division III Athletes (Doctoral dissertation, State University of New York at Binghamton).